UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/627,518	07/28/2000	Gennaro A. Cuomo	RSW9-2000-0090-US1	3351

36736 7590 10/21/2004

DUKE W. YEE
YEE & ASSOCIATES, P.C.
P.O. BOX 802333
DALLAS, TX 75380

EXAMINER	
MAURO JR, THOMAS J	
ART UNIT	**PAPER NUMBER**
2143	

DATE MAILED: 10/21/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/627,518
Filing Date: July 28, 2000
Appellant(s): CUOMO ET AL.

Francis Lammes, Reg. No. 55,353
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed July 2, 2004.

(1) Real Party in Interest

A statement identifying the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

A statement identifying the related appeals and interferences which will directly affect or be directly affected by or have a bearing on the decision in the pending appeal is contained in the brief.

(3) Status of Claims

The statement of the status of the claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Invention

The summary of invention contained in the brief is correct.

(6) Issues

The appellant's statement of the issues in the brief is correct.

(7) Grouping of Claims

The rejection of claims 1-5, 7-15, 17-20, 25 and 26 stand or fall together because appellant's brief does not include a statement that this grouping of claims does not stand or fall together and reasons in support thereof. See 37 CFR 1.192(c)(7).

(8) Claims Appealed

The copy of the appealed claims contained in the Appendix to the brief is correct.

(9) Prior Art of Record

6,470,389	Chung et al.	10-2002
6,470,008	Khuc	10-2002
6,591,250	Johnson et al.	7-2003
6,606,301	Muller et al.	8-2003

(10) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

1. Claims 1, 2, 5 and 7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250).

Regarding claim 1, Chung teaches the invention substantially as claimed, a method in a data processing system for managing a request including a session identification, comprising:

calculating a first value based on the session identification **[Chung -- Col. 4 lines 37-39 – The client IP address which acts as the session identifier has a first value calculated from it by performing a hash function on the session identifier]**;

routing the request to a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server should handle the request]**;

Chung fails to teach determining whether the first server is functional.

Khuc, however, teaches a routing system which determines whether the first server is functional

[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional].

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the teachings of Khuc for determining whether a server is functional or not and routing the request based on that information into the invention of Chung et al. in order to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down. In addition, Chung teaches calculating a second value in response to the first server being non-functional and routing the request to a second server **[Chung -- Col. 7 lines 9-12 – When a server is down, the dispatch will rehash the IP address, i.e. session identifier, and route this and all subsequent packets to the newly mapped server to prevent any lost data packets caused by the failure]**. Chung, however, fails to teach that the second value is calculated based on the first value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the

dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claim 2, Chung-Khuc-Johnson teach the invention substantially as claimed, as aforementioned in claim 1 above, wherein the step of calculating a first value comprises performing a hash function on the session identification **[Chung -- Col. 4 lines 37-39 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Regarding claim 5, Chung-Khuc-Johnson teach the invention substantially as claimed, as aforementioned in claim 1 above, wherein the step of routing the request to a server comprises:

selecting a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value (first value) from the session identifier is compared to a list of servers to determine which server should handle the request]**;

determining whether the first server is functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**; and

routing the request to the first server **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server should handle the request]** in response to the first server being functional **[Khuc -- Col. 9 lines**

32-34 – As long as the percent allocation, as described above, is not "0", the routing systems will route the request to the primary server because it is functional].

Regarding claim 7, Chung-Khuc-Johnson teach the invention substantially as claimed, as aforementioned in claim 5 above, including determining whether the first server is functional using a look-up table **[Khuc -- Figure 5 & Col. 9 lines 37-39 – By comparing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

2. Claims 3 and 4 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389), Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250), as applied to claims 2 and 3 above, respectively, in view of Muller et al. (U.S. 6,606,301).

Regarding claim 3, Chung-Khuc-Johnson teaches the invention substantially as claimed, as aforementioned in claim 2 above, but fail to teach performing a modulus function to form an integer and selecting a server based on the integer.

Muller teaches performing a modulus function on the first value to form a first integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung-Khuc-Johnson, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach selecting a server based upon the integer.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table, as taught by Khuc into the invention of Chung-Khuc-Johnson-Muller in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server.

Regarding claim 4, Chung-Khuc-Johnson-Muller teach the invention substantially as claimed, as aforementioned in claim 3 above, wherein the step of selecting a server comprises looking up the server in a look-up table using the first integer **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**.

3. Claims 8-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008), Johnson et al. (U.S. 6,591,250) and Muller et al. (U.S. 6,606,301).

Regarding claim 8, Chung teaches the invention substantially as claimed, a method in a data processing system for routing a request to one of a number of servers, comprising:

receiving a request including a session identification **[Chung -- Col. 7 lines 64-66 – The request packet contains the client IP address (session identification]**;

performing a hash function on the session identification to form a first hash value **[Chung -- Col. 4 lines 37-39 and Col. 7 lines 64-66 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Chung fails to teach performing a modulus function on a first and second hash value to form a first and second integer.

Muller teaches performing a modulus function on the first or second value to form a first/second

integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach routing the request to a first/second server using the first/second integer in response to the first server being functional.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**, in response to a first server being functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table after determining if the server is functional, as taught by Khuc into the invention of Chung in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server and to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

Furthermore, Chung fails to teach performing a hash function on the first hash value to form a second hash value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claim 9, Chung-Khuc-Johnson-Muller teach the invention substantially as claimed, as aforementioned in claim 8 above, wherein the integer is between zero and the number of servers minus one **[Muller -- Col. 49 lines 52-54 and lines 63-66 – By performing the modulus function over the hash function result using the number of available components, i.e. servers, the resulting integer must be between 0 and N-1 components, i.e. servers]**.

Regarding claim 10, Chung-Khuc-Johnson-Muller teach the invention substantially as claimed, as aforementioned in claim 8 above, wherein the step of routing the request comprises looking up the server in a look-up table using the integer **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that the client should be routed to in order to service the request]**.

4. Claims 11, 12, 15 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250).

Regarding claim 11, Chung teaches the invention substantially as claimed, an apparatus in a data processing system for managing a request including a session identification, comprising:

calculating a first value based on the session identification **[Chung -- Col. 4 lines 37-39 – The client IP address which acts as the session identifier has a first value calculated from it by performing a hash function on the session identifier]**;

routing the request to a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server should handle the request]**;

Chung fails to teach determining whether the first server is functional.

Khuc, however, teaches a routing system which determines whether the first server is functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the teachings of Khuc for determining whether a server is functional or not and routing the request based on that information into the invention of Chung et al. in order to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

In addition, Chung teaches calculating a second value in response to the first server being non-functional and routing the request to a second server **[Chung -- Col. 7 lines 9-12 – When a server is down, the dispatch will rehash the IP address, i.e. session identifier, and route this and all subsequent packets to the newly mapped server to prevent any lost data packets caused by the failure]**. Chung, however, fails to teach that the second value is calculated based on the first value.

Johnson, however, teaches calculating a second value based upon a first value which entails

performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claims 12 and 15, these are apparatus claims corresponding to the methods claimed in claims 2 and 5. They have similar limitations; therefore, claims 12 and 15 are rejected under the same rationale.

Regarding claim 17, this is an apparatus claims corresponding to the method claimed in claim 7. It has similar limitations; therefore, claim 17 is rejected under the same rationale.

5. Claims 13 and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389), Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250), as applied to claims 12 and 13 above, respectively, in view of Muller et al. (U.S. 6,606,301).

Regarding claim 13, Chung-Khuc-Johnson teaches the invention substantially as claimed, as aforementioned in claim 12 above, but fail to teach performing a modulus function to form an integer and selecting a server based on the integer.

Muller teaches performing a modulus function on the first value to form a first integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung-Khuc-Johnson, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach selecting a server based upon the integer.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.
Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table, as taught by Khuc into the invention of Chung-Khuc-Johnson-Muller in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server.

Regarding claim 14, this is an apparatus claims corresponding to the method claimed in claim 4. It has similar limitations; therefore, claim 14 is rejected under the same rationale.

6. Claims 18-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008), Johnson et al. (U.S. 6,591,250) and Muller et al. (U.S. 6,606,301).

Regarding claim 18, Chung teaches the invention substantially as claimed, an apparatus in a data processing system for routing a request to one of a number of servers, comprising:

a processor; and

a memory electrically connected to the processor, the memory having stored therein a program to be executed on the processor **[Chung -- Col. 7 lines 45-47 – By having an operating system (OS), it is inherent that in order for the OS to process tasks and store/run**

not only itself but also the routing algorithm correctly, a processor and memory must be contained within the dispatcher];

receiving a request including a session identification **[Chung -- Col. 7 lines 64-66 – The request packet contains the client IP address (session identification]**;

performing a hash function on the session identification to form a first hash value **[Chung -- Col. 4 lines 37-39 and Col. 7 lines 64-66 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Chung fails to teach performing a modulus function on a first and second hash value to form a first and second integer.

Muller teaches performing a modulus function on the first or second value to form a first/second integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach routing the request to a first/second server using the first/second integer in response to the first server being functional.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**, in response to a first server being functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table after determining if the server is functional, as taught by Khuc into the invention of Chung in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server and to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

Furthermore, Chung fails to teach performing a hash function on the first hash value to form a second hash value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

Regarding claims 19 and 20, these are apparatus claims corresponding to the methods claimed in claims 9 and 10. They have similar limitations; therefore, claims 19 and 20 are rejected under the same rationale.

7. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008) and Johnson et al. (U.S. 6,591,250).

Regarding claim 25, Chung teaches the invention substantially as claimed, a computer program product, in a computer readable medium, for managing a request including a session identification **[Chung -- Col. 7 lines 45-47 - A dispatcher which includes an operating system, i.e. server which inherently contains instructions for carrying out the dispatching tasks]**, comprising:

calculating a first value based on the session identification **[Chung -- Col. 4 lines 37-39 – The client IP address which acts as the session identifier has a first value calculated from it by performing a hash function on the session identifier]**;

routing the request to a first server based on the first value **[Chung -- Col. 4 lines 59-63 – The computed hash value from the session identifier is compared to a list of servers to determine which server should handle the request]**;

Chung fails to teach determining whether the first server is functional.

Khuc, however, teaches a routing system which determines whether the first server is functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to include the teachings of Khuc for determining whether a server is functional or not and routing the request based on that information into the invention of Chung et al. in order to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

In addition, Chung teaches calculating a second value in response to the first server being non-functional and routing the request to a second server **[Chung -- Col. 7 lines 9-12 – When a server is down, the dispatch will rehash the IP address, i.e. session identifier, and route this and all subsequent packets to the newly mapped server to prevent any lost data packets caused by the failure]**. Chung, however, fails to teach that the second value is calculated based on the first value.

Johnson, however, teaches calculating a second value based upon a first value which entails

performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

8. Claim 26 is rejected under 35 U.S.C. 103(a) as being unpatentable over Chung et al. (U.S. 6,470,389) in view of Khuc (U.S. 6,470,008), Johnson et al. (U.S. 6,591,250) and Muller et al. (U.S. 6,606,301).

Regarding claim 26, Chung teaches the invention substantially as claimed, a computer program product, in a computer readable medium, for managing a request including a session identification **[Chung -- Col. 7 lines 45-47 - A dispatcher which includes an operating system, i.e. server which inherently contains instructions for carrying out the dispatching tasks]**, comprising:

receiving a request including a session identification **[Chung -- Col. 7 lines 64-66 – The request packet contains the client IP address (session identification]**;

performing a hash function on the session identification to form a first hash value **[Chung -- Col. 4 lines 37-39 and Col. 7 lines 64-66 – An appropriate hash function is used on the client IP address, also known as, the session identifier]**.

Chung fails to teach performing a modulus function on a first and second hash value to form a first and second integer.

Muller teaches performing a modulus function on the first or second value to form a first/second integer **[Muller -- Col. 49 lines 52-53 – After hashing, a modulus function is performed. This, by the function's nature, produces an integer value]**.

Chung performs a hash function on a session identifier, i.e. IP address, in order to obtain greater speed and efficiency to route requests to a plurality of servers and to prevent traffic and bottlenecks.

Muller, after hashing a value, further performs a modulus function to further provide more speed and efficiency in distributing the processing requests and to prevent "bottlenecks" **[Muller -- Col. 49 lines 52-54]**.

It would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate performing a modulus function on a hash value, as taught by Muller into the routing system of Chung, in order to further provide increased speed and efficiency in routing requests and to prevent bottlenecks from occurring.

In addition, Chung fails to teach routing the request to a first/second server using the first/second integer in response to the first server being functional.

Khuc teaches a look-up table which uses integers as the index into the table to determine which server to route requests to **[Khuc -- Figure 5 & Col. 9 line 10 – The first (indexing) column comprised of integer values in the look-up table is used to select the IP address of a particular server that should service the request]**, in response to a first server being functional **[Khuc -- Col. 9 lines 36-39 – By changing the percent allocation field in the look-up table, this will determine whether the server is function, i.e. percent allocation = 0, then non-functional and percent allocation >0, functional]**.

Both Chung and Khuc are both interested in quickly and efficiently routing requests to servers. Khuc uses a look-up table to further speed up the routing process.

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the routing of request using integers in a look-up table after determining if the server is functional, as taught by Khuc into the invention of Chung in order to further provide a fast and efficient method for quickly finding and routing requests to the correct server and to provide a level of fault-tolerance to prevent the dispatching of a client to a server which is down.

Furthermore, Chung fails to teach performing a hash function on the first hash value to form a second hash value.

Johnson, however, teaches calculating a second value based upon a first value which entails performing a hash function on a first value already calculated from a first hash, i.e. H (K, H (K, M)) **[Johnson -- Col. 13 lines 30-32]**.

Chung discloses that a second value is calculated when a server is non-functional, i.e. an attack has caused a server to fail.

Johnson performs this second hash to further provide security such that attacks are prevented. It would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the calculating of a second value based upon a first value, i.e. hash functions, as taught by Johnson into the system of Chung, in order to achieve greater security in the dispatching system which further masks which server requests are being routed through to prevent another attack from being staged on one of the remaining functional servers.

(11) Response to Argument

(A) Applicant contends that the combination of Chung and Johnson fails to teach calculating a second value based on the first value in response to the first server being non-functional, whereas claims 1, 11 and 25 call for this limitation.

In response to argument (A), Examiner acknowledges the diagrams provided which intend to show the various flow diagrams for both the prior art and the instant invention, however, respectfully disagrees with the interpretation and diagram depicting the combination of Chung and Johnson. Johnson discloses calculating a second hash value from a first hash value that has already been hashed previously (See Johnson Col. 13 lines 30-32). Chung discloses that a first hash value is calculated from the client IP

address and is then routed to the mapped server. See Chung Col. 4 lines 37-39 and 59-63 along with Col. 6 lines 62-67 and Col. 7 lines 64-67 – Col. 8 lines 1-3. Chung further teaches that after routing the packet, "when a server in the cluster fails, the subset of clients assigned to that server will not be able to connect to it." See Chung Col. 7 lines 5-6. Thus already the diagram of Chung-Johnson is incorrect as the "**Server 1 Not Operational**" block should be indicated in exactly the same way as that shown in the presently claimed invention diagram. It is found in the Chung reference that if a first server is unavailable, i.e. non-functional, a second value is calculated. See Chung Col. 7 lines 9-12. The combination, as is interpreted by the Examiner, brings in Johnson to rehash the already hashed value in Chung only after the first server is deemed to be non-functional (See final rejection above), not immediately after the first hash value is calculated, as is asserted in the diagram of the combination of Chung and Johnson submitted by the Applicant. Nowhere is it found in Johnson that the second hash value is calculated immediately after the first hash value is calculated, as is contended by the Applicant. Thus, Examiner disagrees with the Applicant's argument and reiterates that the Johnson reference, namely calculating a second hash value from a first hash value, is brought in only after the server has been determined to be non-functional, as was shown in Chung, thus teaching the exact same diagram as is claimed by the present invention.

In conclusion, Applicant (See Page 9 of the Appeal Brief) agrees that Chung calculates a second hash function if a first server is unavailable; however, the second hash function is not calculated based on the first value from the first hash function. Johnson provides the

deficiency in calculating a second hash value from an already hashed value, thus arriving at the applicant's invention.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

TJM
October 12, 2004

Conferees
David Wiley

Jack Harvey

JACK B. HARVEY
SUPERVISORY PATENT EXAMINER

William Vaughn

Primary Examiner
Art Unit 2143

A Bruce Clay
IBM Corporation Dept T81 062
3039 Cornwallis Road
Research Triangle Park, NC 27709